

Blood Bank Management System

Nilasini Thirunavukkarasu

Department of Computer science and engineering, University of Moratuwa, Sri Lanka
Nilathiru.12@cse.mrt.ac.lk

Abstract—“BLOOD” is one of the most essential necessities of our lives. Mostly during accidents or most of the hospitals issues the need for blood is the common problem. In addition to emergency requirements, advances in medicine have increased the need for blood in many ongoing treatments and elective surgeries. This paper presents an android application blood bank management system that enables individuals who want to donate blood to help the needy. It also enables hospitals to send blood request to the donors who were already registered to the system. The system has developed by using java, PHP, and MySQL as a database system to manage and store the data. The Waterfall Methodology, which is the traditional version and the classic approach of a system development life cycle, is used to develop and build the android based blood bank system. The system targets three types of user: the public who wants to donate blood, the hospitals who need blood, and the admin who manage the system. In this application the system has utilized the Google distance API to find the distance between donors that are in favor of donating blood and hospitals. The android application is readily scalable, efficient and adaptable to meet the complex need of blood for the hospitals.

Keywords—Blood Donors; Mange Blood Bank .

VII. INTRODUCTION

Blood bank is a provider of blood services meeting the needs of donors, patients, and the wider healthcare community. The application is a blood donor search and notification application, which is named as bbmSystem. Blood is essential to everyone. Mostly in the rural areas getting blood within the specific time is difficult because of the lack facilities in maintaining a proper management of blood donor’s details.

The main objective of this Blood Bank Management system (BBM) is to find the nearest blood donors. It allows blood donors and hospitals to register to the system by giving the relevant details. Whenever a hospital needs blood they can find out the nearest donors and send blood request through the system. Then blood donors can visit the hospital to donate blood. In addition that this application requires the storage of blood donor details. Blood donor details have been stored in the MySQL database which is hosted in the cloud in order to provision the required resources. To communicate android to MySQL database some web services need to be deploy since there are no direct approach available. So the require web services implemented in PHP to expose the database operations. To deploy MYSQl database and PHP server “openshift” has been chosen as it provides quick solution by using github.

Internet connectivity is important for the proper function of this system. It will not be a problem, as most of the android users are always connected to internet to experience the real usage of the smart phones. Although most people have smart phones some of them might not aware of how to handle bbmsystem. In order to assist them a help page is available to get familiar on how to operate this BBM System. The assumption is considered for this product as the users will require having a smart phone which supports android platform. In addition to that, the phone will also have facility of location finder. This application is fully dependent on the internet connection of the users.

The paper is structured as follows: Section II discusses related background of this application. Section III describes the requirements and design aspects of the BBM application. The implementation considerations and Graphical User Interfaces (GUIs) explain in Section IV. Section V states the testing and analysis of the system and Section VI concludes the paper stating future possible extensions for the system.

VIII. LITERATURE REVIEW

Nowadays blood demand has increased. Even there is sufficient blood donors in the society there is a lack in blood because there is no proper management for blood and the existing blood management websites are not updated correctly. During the proposal phase of this project, a research has been carried out against on the similar existing applications. Research revealed that there are many similar applications in the Google play market, claiming that they provide blood donor details but after downloading many of those, it is found that many of those applications contain some sort of faults and applications doesn’t contain enough registered blood donors because many of the people seems to be not interested in displaying their personal details to public, the existing blood bank applications do not consider the medical conditions about blood donors when donors registering and there is no application to find the nearest donor [8, 9]. The existing applications only send the notifications to all blood donors who have the specific blood group.

It is found that there is no single application in the market which is mainly focused on Sri Lankan blood donors [1]. Blood bank system evolved to meet the Sri Lanka’s blood needs and to solve above mentioned within Sri Lankan context.

IX. SYSTEM MODELS

A. System requirement

The core requirements of the application is that blood donors should be able to register to the system and hospitals should be able to request the specific type of blood, notifications should be sending to the selected list of blood donors. Fig. 1, shows the usecase diagram of this system.

The non-functional requirements satisfied by this application are usability, availability and performance. Usability is achieved by the good GUI based on the user comments. Availability and performance are achieved by proper testing and the hosting of the database in a trusted server.

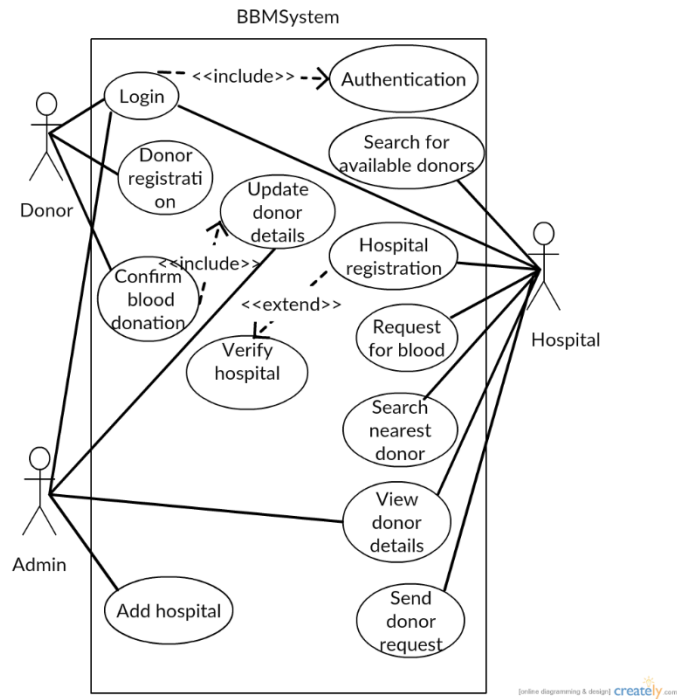


Fig.1. Usecase diagram

In order to get the notifications about blood emergency, blood donors are requested to register to the system by giving the details such as first name, age, blood group, available location, contact number and availability. After registering to the system, system will save those details in a database and each blood donors are uniquely identified by their contact number. In addition to that each blood donors are requested to select a user name and password for their future logins. Blood donors are expected to update the details themselves continuously, in order to keep the details up to date.

In order to make request for donors, the hospitals must already register to the system. When they are registering, the hospitals need to be in the authorized hospitals list which will be created by admin. If the hospital name is not in the authorized list then the hospital must verify their hospitals by sending a SMS to the admin. After that admin will clarify and add the hospital as an authorized hospital.

During a blood emergency, hospitals are expected to access this application and send a request for a blood group. Hospitals are expected to the required blood group and the hospital

location as the input. Blood group is essential to locate the possible blood donors for the particular blood emergency and hospital location is required to find the nearest blood donors within the plenty of available donors from various part of the country. After that the system will send SMS notification to the identified donors. Upon the notification, the donors can confirm their donation and can visit to the specified location to donate blood. When blood donor confirms his/her donation for blood request, he/she has to update their information on the system of the further availability of blood donation as most of the time blood donation is restricted to six months. So the donor must update their details while confirming their donation.

B. System design

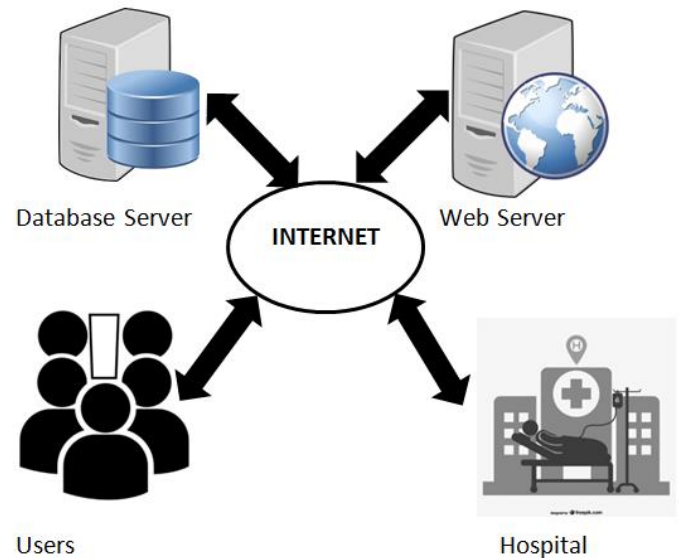


Fig.2. High level picture

Fig. 2, depicts the high level picture of the architecture of application. Web server does the necessary task according to the requests. When hospitals send blood request to the nearest donor, the distance between each donor location and hospital location will be found by using the Google distance Application Programming Interface (API). Then the distances will be sorted using Collections API. After that minimum distance location of the donors will be selected and shown as the nearest donors. After that notification will be sent to the selected users using the SMS manager facility of the android.

Model-View-Controller architecture is used for the design of the application architecture, since it can be easily handled in Android. MVC supports flexibility by separation of responsibilities. Fig. 3, describes the overview of the MVC architecture of the application.

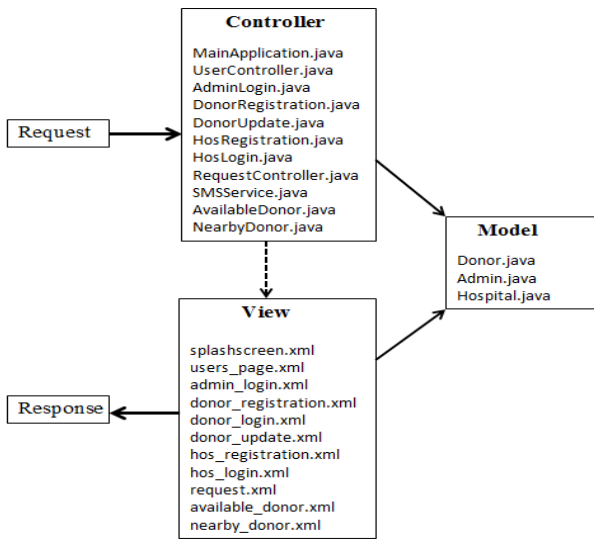


Fig.3. MVC architecture

In this model the application is divided into 3 separate components called model, view and controller. They are interconnected. Controller controls overall application. View gets the support from model and controller in order to output to the user. Model is used to save the temporary data of the application. Fig. 4, describes the core part of the class diagram for the system.

Fig. 5 shows the activity diagram for the main control flow of the BBM system. Hospitals that need blood can request blood through this application indicating blood group needed and the hospital or hospital to provide blood. After that this application will find out the nearest blood donors with specified blood group using location and notify them by using SMS. After get notified, the donors confirmed their donation.

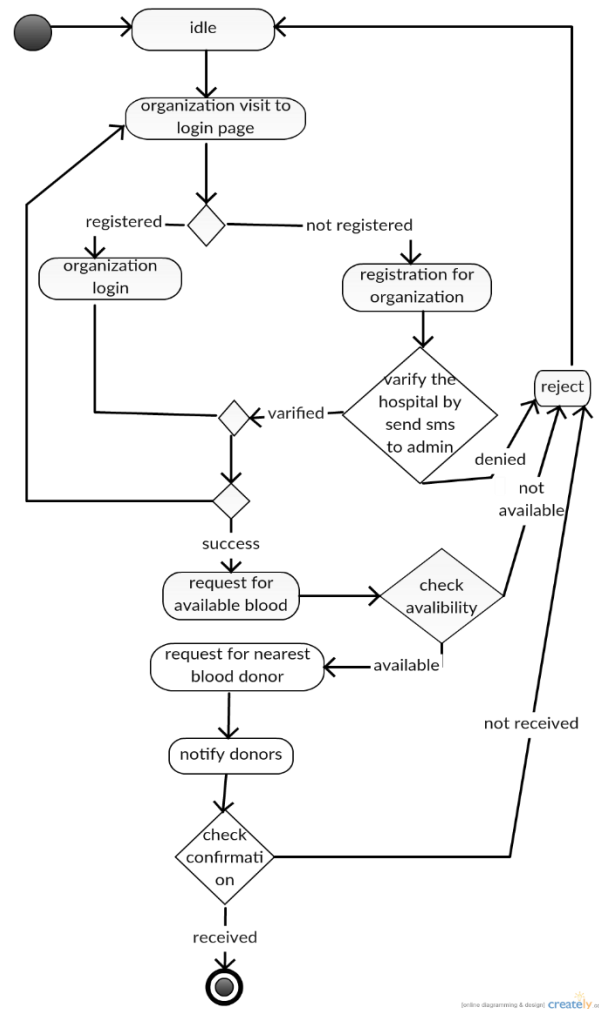


Fig.5. Activity diagram

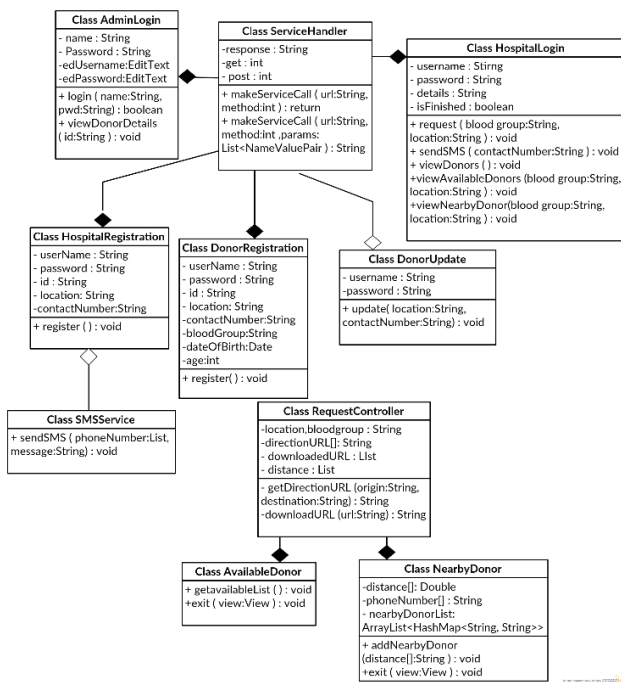


Fig.4. Class diagram of the system

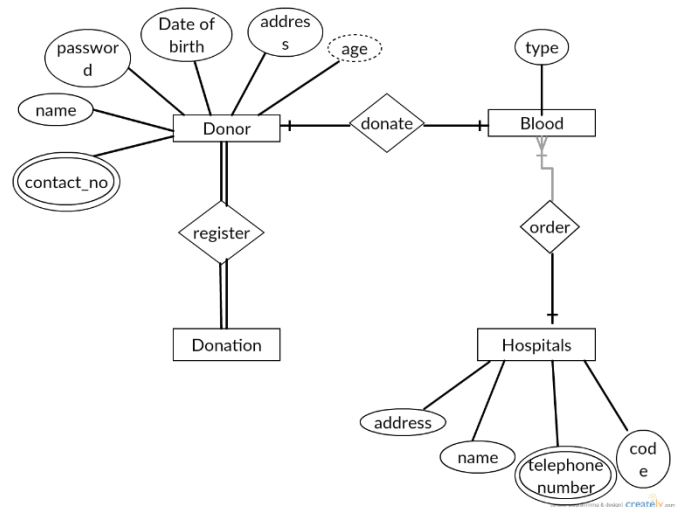


Fig.6. ER diagram

C. System Database

Database is designed for manage donor details, hospital details and authorized hospital details. Donors are stored in the donor table. It contains blood group, username, password, location, name, contact number, and age as the attributes. It uses contact number as the primary key. Hospitals are stored in hospital table. It contains username, password, bbmcode, hospital name, contact number and email address as the attributes. It uses bbmcode as the primary key. Authorized hospitals are stored in the hospital table. It contains hospital name and location as the attributes.

X. SYSTEM IMPLEMENTATION

A. Implementation procedure

The BBM system is implemented using the java, PHP and XML languages. Java is used as the main implementation language. To communicate android to MYSQL database some web services need to be deployed since there are no direct approach available. So the required web services implemented in PHP to expose the database operations. To deploy MYSQL database and PHP server “openshift” has been chosen as it provides quick solution by using GitHub. When using “openshift” we only need to deploy our files in the GitHub and give the location in the “openshift”. The pseudocode in Fig. 7 shows, how to find the available donor details by a hospital.

```

1) User login (hospital staff)
2) If login successful
   b_group ← user input
   donor_details json array ← retrieve from db
3) For array length
   Get the json object (row)
   Get every donor column
   Search for the particular blood group in the donor
   database
   If matching blood group found
   Store every donor detail with blood group
   If status is available
   directionsurlList ← Get the directionURL
   between the locations
   Donor_detailHashMap ← dd the donor details in to
   a hash map
   Available_donorlist ← dd Donor_detailHashMap
4) While previous asyntask finished
   Sleep
5) Connect with url
   directionList ← download the specific url for
   every directionsURL
6) While previous asyntask finished
   Sleep
7) jsonObject ← convert directionList using JSON
   parser
   Distance ← convert between objects and json
   arrays up to get the distance

```

Fig.7. Pseudocode for searching donor details

Since this solution only satisfies demo and development purposes due to the limited resource available for free in openshift. The system has to be deployed in amazon or in

another proper infrastructure in order to scale the system in long term. PHP scripts is used to run the server end jobs such as registering, logging in , getting the available blood donors , getting the nearest blood donors and sending notifications.

Eclipse is used as the IDE for the system implementation. Google direction API is used to find the distance between two specific locations. When we try to get the distance between two locations, the result will get as string with JavaScript Object Notation (JSON) format. Therefore a JSON parser is used to convert this in to object and a JSON ARRAY is used to convert the object in to JSON array. This was used to split the JSON format string and get only the distance because the result from direction API contains more details additional to the distance.

B. Main Interfaces

Fig. 8 – Fig. 13 show the GUIs of blood Bank Management system. From the main interface, different users are directed to relevant interfaces based on their role.



Fig.8.Main menu for all users

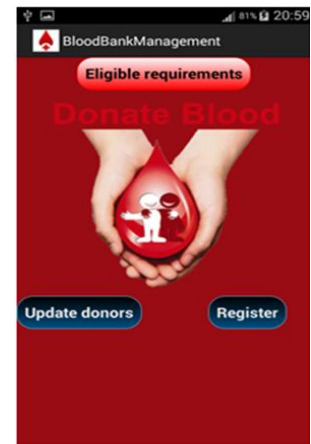


Fig.9.Main menu for donors

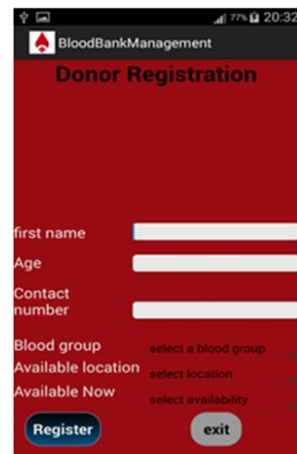


Fig.10. Donor Registration

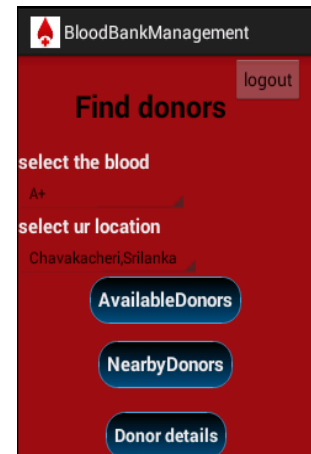


Fig.11. Find Donors

As shown in Fig. 9, the system has an option to view eligibility requirements for a blood donor. This interface facilitates to register and update details of a donor. A person who is willing to donate blood has to register in the system by entering his/ her details as given by Fig. 10. As shown in Fig.

11, the application facilitates to search blood donors for a given blood group and a location. Once the system shows the available donors (Fig. 12) with their details, the user can filter nearby donors. When the system shows the details of nearby donors (Fig. 13) a send request can be send to a selected donor via sms.

VI. CONCLUSION AND FUTURE WORK

This paper describes the design and implementation details of a blood bank management. This system mainly support with functionalities such as donor registration, hospital registration, blood and hospital login, sending blood needed request notification, finding nearby donor details and sending SMS to the admin to approve the hospitals. This application is developed using android and PHP.

This application can be extended to track user location using Google API, to make the application more usable. Currently the hospitals only can find the nearby donors while they can not specify their required distance, the search function can be improved in a way that a user can ask their required distance and they can find the donors within that distance circle. The database server is hosted in "open Shift" and expecting to publish this application in Google play market. In order to validate the application an empirical study can be carried out within medical organizations in Sri Lanka. This application is based on social service without profitability purpose.

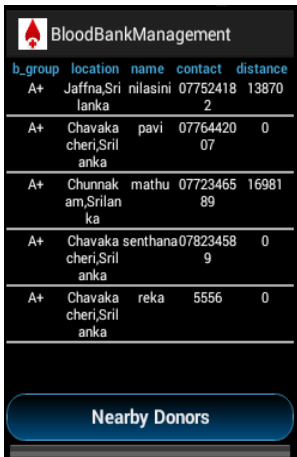
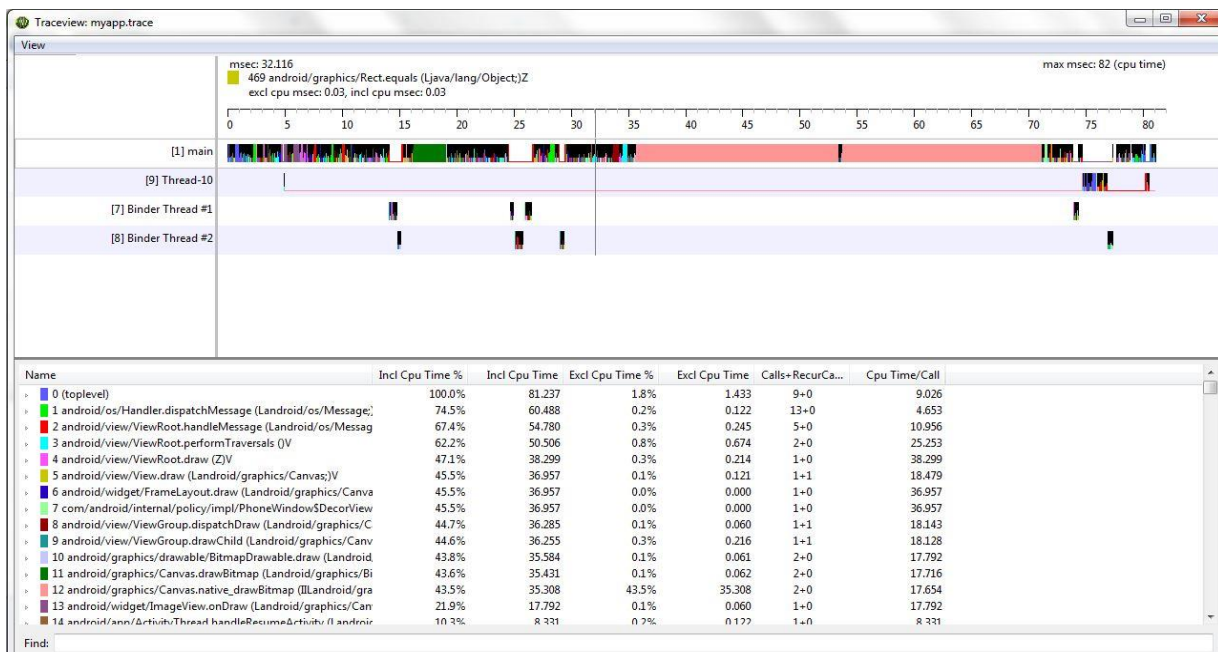


Fig.12.Available donors



Fig.13.Nearby donors



V. SYSTEM TESTING AND ANALYSIS

Unit testing has done using JUnit testing mechanism and integration testing is done using the Robotium library. Test classes are used for testing the First Screen Activity. It has checked whether the components are not null and the expected values and the actual values are same. Integration testing is done by giving the input details programmatically and checking the activities. It has programmatically input the registration details and checks the login, logout activities. The performance of the application using Trace view is shown in Fig. 14.

REFEERNCES

- [1] Play store, "Blood bank applications", [online]. Available: <https://play.google.com/store/search?q=blood%20bank%20management&c=apps&hl=en>
- [2] Wikipedia contributors, "Rational Unified Process", [online]. Available: https://en.wikipedia.org/wiki/Rational_Unified_Process
- [3] Vogella.com, "Robotium testing", [online]. Availbale: <http://www.vogella.com/tutorials/Robotium/article.html>
- [4] Lifesciencesite.com, "existing blood reservation systems", [online].Available http://www.lifesciencesite.com/ljsj/life1108/008_24337life110814_60_65.pdf
- [5]Ajer.org, "existing blood reservation systems", [online].Available [http://www.ajer.org/papers/v3\(2\)/O032105108.pdf](http://www.ajer.org/papers/v3(2)/O032105108.pdf)
- [6]

Fig.14. Performance of the Application using Trace View

- Ijera.com, "Blood bank management information system" [online].Available <http://www.ijera.com/papers/vol%201%20issue%202/012260263AF.pdf>
- [7]"Blood bank management system using cloud computing",[online].Available: <http://researchtrend.net/ijet21/ijetnew/5%20JAVED%20AKHTAR%20KHAN.pdf>.
- [8] playstore, "existing blood bank applications",[online].Available: <https://play.google.com/store/apps/details?id=com.startsmall.mybloodbank&hl=en>
- [9] playstore, "existing blood bank applications",[online].Available: <https://play.google.com/store/apps/details?id=com.cube.arc.blood&hl=en>